Implementing universal dependency, morphology, and multiword expression annotation standards for Turkish language processing

Sulubacak, Umut

2018-05-30

Sulubacak , U 2018 , ' Implementing universal dependency, morphology, and multiword expression annotation standards for Turkish language processing ' , Turkish Journal of Electrical Engineering and Computer Sciences , vol. 26 , no. 3 , 43 , pp. 1662-1672 . https://doi.org/10.3906/elk-1706

http://hdl.handle.net/10138/237336
https://doi.org/10.3906/elk-1706-81

unspecified
publishedVersion

Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2018) 26: 1662 – 1672

doi:10.3906/elk-1706-81

Implementing universal dependency, morphology, and multiword expression annotation standards for Turkish language processing

Umut SULUBACAK, Gülşen ERYİĞİT*
Department of Computer Engineering, Faculty of Engineering, İstanbul Technical University, İstanbul, Turkey

Received: 07.06.2017 • **Accepted/Published Online:** 09.01.2017 • **Final Version:** 30.05.2018

Abstract: Released only a year ago as the outputs of a research project ("Parsing Web 2.0 Sentences", supported in part by a TÜBİTAK 1001 grant (No. 112E276) and a part of the ICT COST Action PARSEME (IC1207)), IMST and IWT are currently the most comprehensive Turkish dependency treebanks in the literature. This article introduces the final states of our treebanks, as well as a newly integrated hierarchical categorization of the multiheaded dependencies and their organization in an exclusive deep dependency layer in the treebanks. It also presents the adaptation of recent studies on standardizing multiword expression and named entity annotation schemes for the Turkish language and integration of benchmark annotations into the dependency layers of our treebanks and the mapping of the treebanks to the latest Universal Dependencies (v2.0) standard, ensuring further compliance with rising universal annotation trends. In addition to significantly boosting the universal recognition of Turkish treebanks, our recent efforts have shown an improvement in their syntactic parsing performance (up to 77.8%/82.8% LAS and 84.0%/87.9% UAS for IMST/IWT, respectively). The final states of the treebanks are expected to be more suited to different natural language processing tasks, such as named entity recognition, multiword expression detection, transfer-based machine translation, semantic parsing, and semantic role labeling.

Key words: Turkish, treebanks, natural language processing, dependency parsing, deep dependencies, multiword expressions, universal dependencies

1. Introduction

The field of natural language processing (NLP) has attracted massive interest for decades. Until now, it has been applied to numerous languages and a variety of disciplines and has branched out to many specializations. With the integration of machine learning and supplementary methodologies to elevate efficiency and access to high volumes of data through the Internet, NLP has become the subject matter in products that enable people to interact with them using natural language. However, high-level applications rely on semantic values that cannot be extracted from raw transcripts, requiring linguistic resources to be formally described in ways that would enable them to be worked through computational methods.

Language processing tools (e.g., sentence splitters, tokenizers, morphological analyzers, syntactic, and semantic parsers) are often run consecutively in a pipeline, and they each make a recurring appearance as a preprocessing component for various language applications. For this reason, such low-level processing tasks are often the most studied ones and also the most challenging, considering their critical contribution to the performance of higher-level systems.

*Correspondence: gulsen.cebiroglu@itu.edu.tr

The application of supervised machine learning techniques to the tasks of morphological tagging and syntactic parsing requires gold-standard training corpora that are tokenized and morphologically and syntactically annotated by hand, called treebanks. When dependency formalism [1–3] is used to represent syntax in a treebank, it is called a dependency treebank. Consistent annotation and a diverse composition define the measure of quality for dependency treebanks and are of paramount importance.

Until quite recently, the METU-Sabancı Turkish Treebank [4,5] was the only dependency treebank for Turkish sentences that was well edited and large enough for general use, and it has been utilized and evaluated in many studies [6–8]. The ITU-METU-Sabancı Treebank (IMST) [9] was later developed as the output of our research project [10], as a reannotated version of the METU-Sabancı Treebank, following a revised annotation framework. IMST proved to be a robust resource [9], despite being a relatively young treebank [11]. Another new resource is the ITU Web Treebank (IWT) [12], which is the first Turkish web treebank and one of the first fully annotated treebanks of user-generated content worldwide, following its international predecessors, the Google English Web Treebank [13] and the French Social Media Bank [14].

The development of IMST and IWT was not discontinued after their initial release, as we continued to maintain and improve them, ensuring that they remained the state of the art among Turkish language resources. We have shown in previous studies [9,12,15] that the treebanks are ready to tackle computational challenges, contend with their international counterparts, and keep up with the universal standards in corpus development. This article introduces the final states of these treebanks and the first empirical parsing results on them (Section 4.2) with the use of a data-driven dependency parser, as well as the newly added features listed below:

- A hierarchical categorization of overlapping dependencies in an independent deep dependency layer,
- Integration of the most recent benchmarks in multiword expression and named entity annotation into the dependency layer,
- Compliance with the latest Universal Dependencies standard (UD v2.0).

The article is structured as follows: Section 2 provides some preliminary information on morphological analysis and dependency formalism and then discusses the properties of the Turkish language in relation to language processing. Section 3 summarizes the progression of IMST and IWT and then outlines the contributions we made in order to advance their development. In Section 4, we present statistics from the final versions of the treebanks before moving on to describe our empirical evaluations of them, along with a discussion of the resulting figures. Finally, we present our conclusion in Section 5.

2. Dependency parsing of Turkish

The last decade brought about the rise of dependency parsing in syntactic parsing as a formalism that is well suited to supervised machine learning methodologies [6]. Establishing dependency grammar is a challenging problem for Turkish, which makes a compelling case for linguistic studies with its characteristic agglutinative typology, extreme morphosyntactic derivation capabilities, and abundance of ambiguous cases.

Many studies have been conducted on the morphosyntactic analysis of Turkish since the early 1990s. However, research groups have only recently started to focus on analyzing varieties in noncanonical language and developing sophisticated language resources to utilize in machine learning systems. While this trend facilitates the creation of language processing applications for Turkish that were previously impossible, it also portends that Turkish will eventually be on par with other well-studied languages.

3. New language resources

IMST and IWT were created as implementations of a new annotation framework that was tailored to address specific issues in tokenization, domain limitations, morphological disambiguation, and syntactic parsing. The treebanks were also designed to be flexible in accommodating future studies and to be original in different aspects, as well.

The initial releases of the treebanks featured a multiheaded annotation scheme, i.e. where the annotation allowed dependents to be assigned more than one head token in order to formally represent ambiguity in syntactic analysis. However, this representation was incompatible with the current annotation conventions for secondary dependencies [16]. Our current releases of IMST and IWT feature a separate, manually annotated deep dependency layer, effectively distinguishing between primary and secondary heads.

In addition to syntactic dependencies, we also integrated multiword expression (MWE) annotations into the dependency layers of the treebanks. Our first rendition of the MWE annotations also included named entities, but it was decidedly rather primitive. In accordance with more recent studies that aimed to establish annotation standards for MWEs and named entities [11,17,18], we supplied the current releases of IMST and IWT with more comprehensive and systematic MWE annotations, all manually annotated.

IMST was recently semiautomatically converted in compliance with the universal standards of tokenization and morphological and syntactic annotation as set forth by the UD initiative [19]. The converted IMST–UD Treebank [15] has since been separately maintained and enhanced with more detailed annotations. A UD-compliant mapping of IWT is also underway for the next release. With these, the state of the art in Turkish treebanks has attained a universally recognized composition. We provide detailed descriptions of our contributions to these aspects in the following subsections.

4. Deep dependency layers

Even though dependency annotation often requires a single head token to be specified for each dependent, the relations between the tokens of a sentence can be more intricate than this shallow representation can express. To alleviate this restriction, treebanks may be augmented with an additional layer of deep dependencies in order to represent implied semantic relations. Featured in the fully annotated sentence example in Figure 1, deep dependencies are secondary dependencies from a token to other implicit heads in addition to its surface (primary) head as in the shallow representation. Without such a representation, it is sometimes not possible and otherwise computationally complex [16] to determine the relations denoted by deep dependencies, as demonstrated in Figure 2.

A typical example for the application of deep dependencies is seen in the shared modifiers of the conjuncts in a coordination structure. While shared modifiers theoretically modify each of the conjuncts, the shallow representation forces them to depend on only one. Deep dependencies are also utilized in a number of other cases. Raised subjects cause ambiguity in the syntactic head for the corresponding subject dependency, which is rectified by the usage of deep dependencies, as shown in Figure 3. Relative clauses, constructed via participles in Turkish, constitute several syntactic cases that require deep dependencies to represent implied meanings, as shown in Figure 4. Since these are all common discourse tools, it is common to encounter several cases that call for deep relations within a single sentence, as seen in Figure 5. As such, the annotation of deep dependencies comprises a layer that provides the additional semantic expressiveness that enables the application of the treebanks to the task of semantic role labeling.

The multiheaded representation previously used in IMST and IWT was identical to a deep dependency

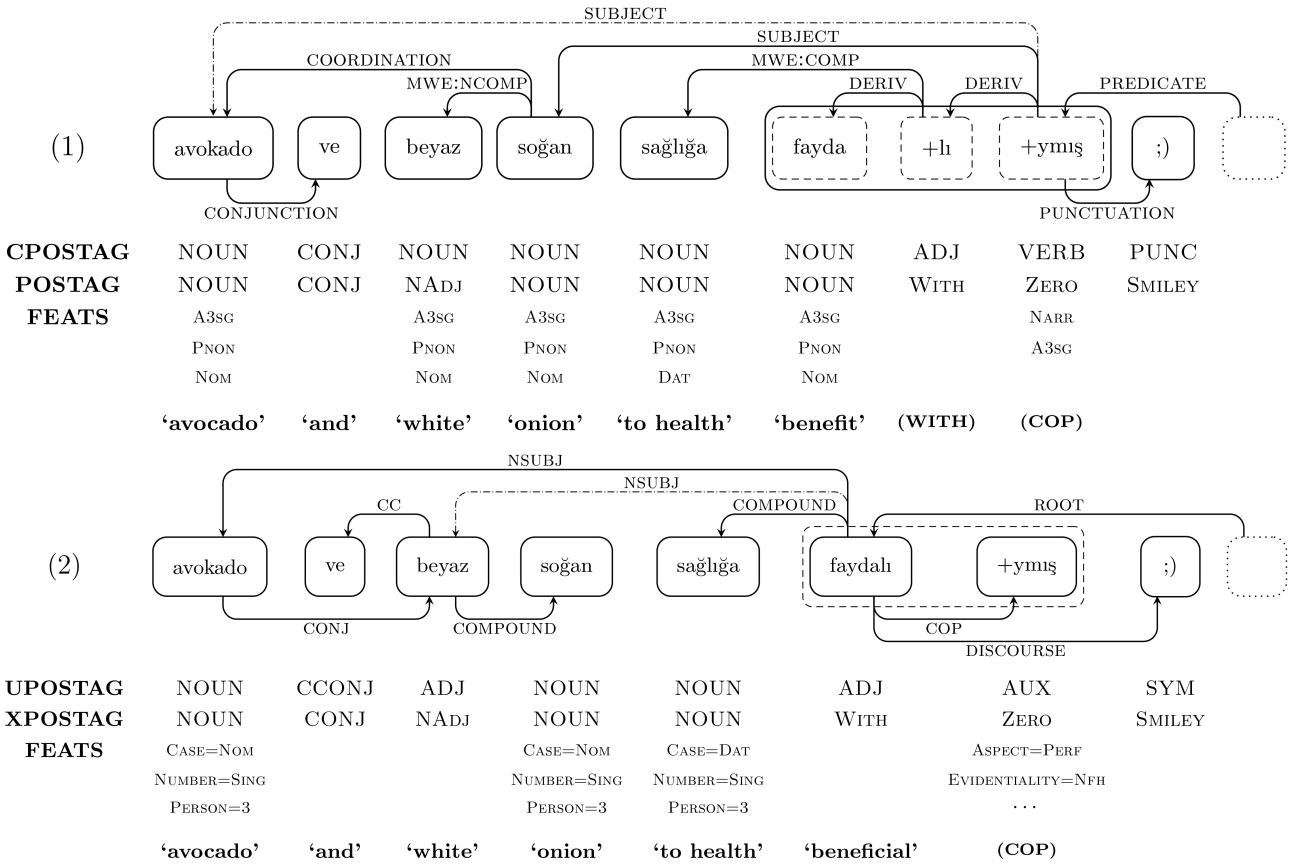

Figure 1. Example of a sentence: "*avokado ve beyaz soğan sağlığa faydalıymış*" ("[I heard] avocado and white onion is healthy"), tokenized and morphosyntactically annotated after the 1) original and the 2) Universal Dependencies frameworks. Dash-dotted arcs represent deep dependencies.

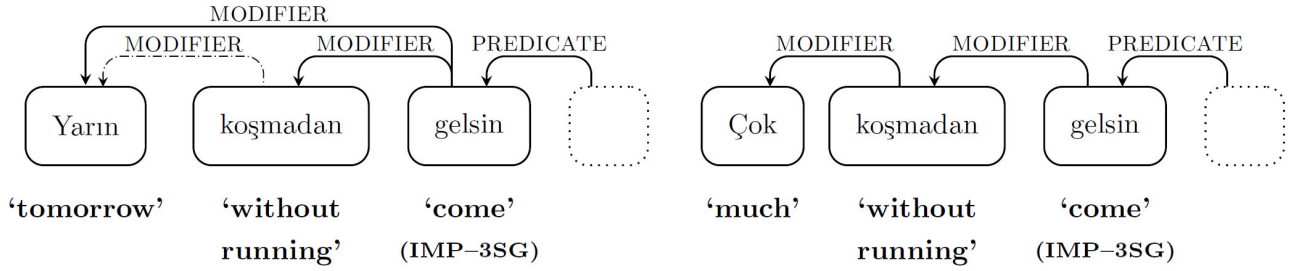

Figure 2. Examples of deep dependencies present and absent in two syntactically analogous clauses: "*Yarın koşmadan gelsin*" ("[Let her/him] come without running tomorrow") and "*Çok koşmadan gelsin*" ("[Let her/him] come without running much").

layer except for the fact that it lacked a hierarchy between surface and deep heads. This initial representation primarily aimed to support a relaxed evaluation metric for the predicted dependencies of a multiheaded token, which could validate any prediction as long as it corresponded to one of the annotated gold-standard heads. However, the representation lacked a gold-standard annotation for surface heads (as also required by universal standards) and necessitated the use of head selection heuristics [20].

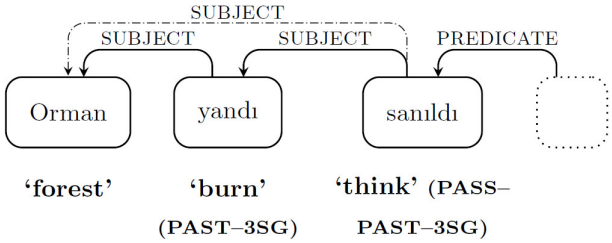

Figure 3. Example of deep dependency annotation used to represent semantics in raised subjects: "*Orman yandı sanıldı*" ("The forest was thought to have burned").

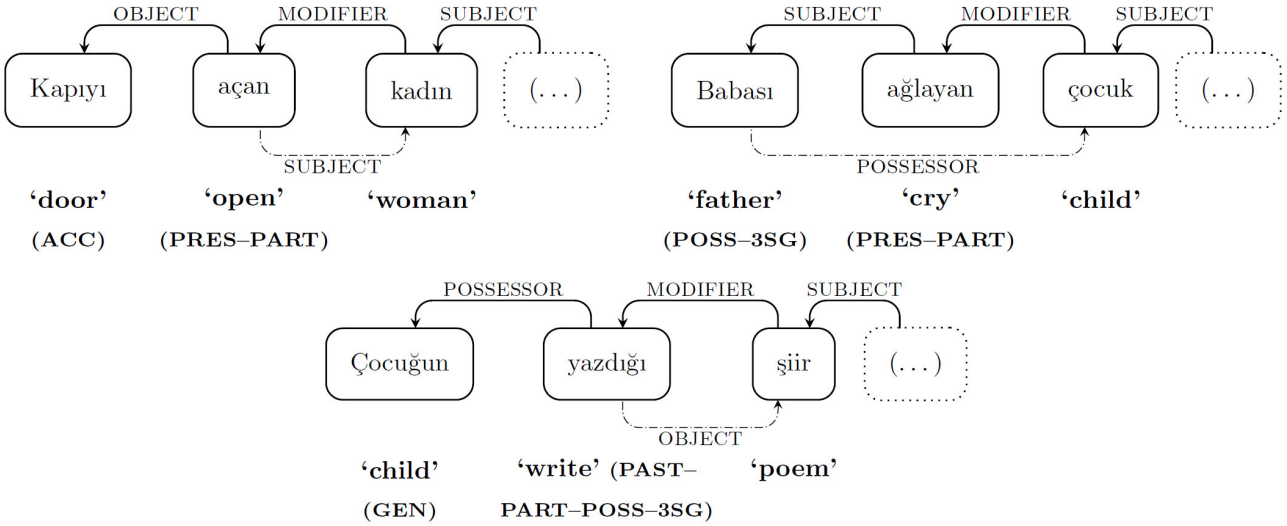

Figure 4. Examples of implied semantic links represented by deep dependencies as a subject in "*Kapıyı açan kadın*" ("The woman who opens the door"), a possessor in "*Babası ağlayan çocuk*" ("The child whose father is crying"), and an object in "*Çocuğun yazdığı şiir*" ("The poem that the child wrote"), depending on a token in a relative clause.

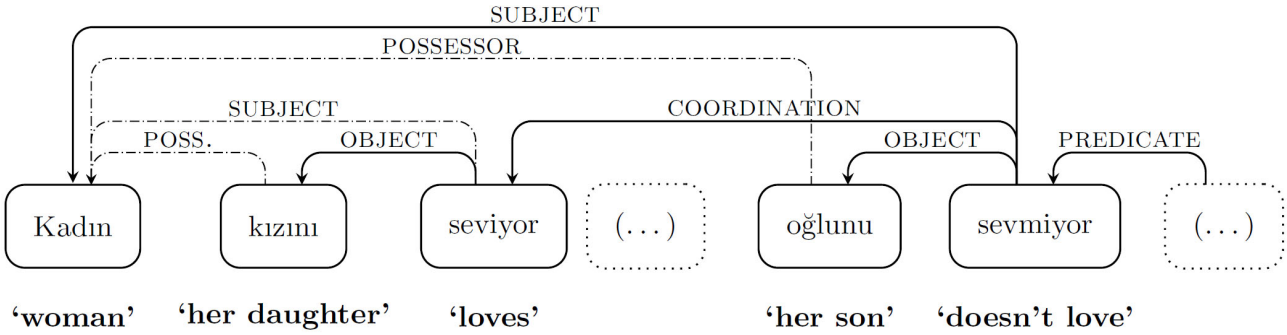

Figure 5. Examples of three deep dependencies in the same sentence: "*Kadın kızını seviyor... oğlunu sevmiyor*" ("The woman loves her daughter... doesn't love her son"), covering an elided reflexive pronoun and two shared modifiers (intra- and interclause).

Our first contribution involved a thorough process of manually selecting surface heads for multiheaded tokens and for the extraction of single-headed representations for both treebanks. We later merged the two representations, moving secondary heads into the deep dependency layer (i.e. an extra field in the CoNLL

dependency representation scheme [6]). In the final outcome, both IMST and IWT were provided with the abstraction of a gold-standard deep dependency layer, eliminating the need for heuristic surface head selection.

4.1. Enriched multiword expressions

Another advantage of IMST and IWT over their predecessors is the annotation of MWEs in the dependency layer. MWEs are lexical items that can be decomposed into multiple lexemes. They display lexical, syntactic, semantic, pragmatic, and/or statistical idiomaticity [21]. The practice of annotating a layer of MWEs on top of underlying syntactic relations is favored due to the expressive power it provides, so it has been employed in a large number of treebanks [17].

Named entity annotation has likewise attracted a great deal of attention. Named entities are labels for one or more successive tokens that denominate a specific person, organization, location, number, or time. In this study, we are only concerned with multiword named entities due to their syntactic value. Since such named entities are structurally similar to MWEs, they were treated as a subcategory. Nonetheless, they exhibit particular properties and specific problems in both annotation and classification [18,22].

Currently, there is little agreement on the universal standards for how MWEs should be annotated [17], though earlier studies proposed elementary approaches for these tasks for the Turkish language [23–25]. Due to a lack of standard annotation schemes, IMST and IWT initially contained a limited set of basic MWE annotations (discourse markers, named entities, and some verbal constructions). After the treebanks were published, the ICT COST Action PARSEME [26] and recent workshops on multiword expressions made significant progress in establishing a basis for multiword expression analysis. Furthermore, some recent studies laid out stronger foundations for MWE analysis in Turkish [11,27] and implemented their approaches in separate annotation layers.

Building on these concerted efforts, we augmented the syntactic annotation framework of IMST and IWT with well-defined schemes for MWEs as part of this study. This work was followed by the integration of the supplementary manual annotation layers directly into the dependency layers of the treebanks. The resulting enhanced dependency layer contains the fine-grained MWE dependency types summarized in Table 1.

The annotated MWEs include expressions such as MWE:FORMEX, e.g., "*iyi geceler*" ("**good night**"); MWE:IDEX, e.g., "*umudu kesmek*" ("**give up [on]**", lit. "**cut hope**"), and MWE:SIMEX, e.g., "*dev gibi*" ("**huge**", lit. "**giant-like**"). They also include compounds such as MWE:COMP, e.g., "*bir şey*" ("**something**", lit. "**a thing**"); MWE:CONJ, e.g., "*ya da*" ("**or**"); MWE:DUP, e.g., "*bir bir*" ("**one by one**", lit. "**one one**"); MWE:LVC, e.g., "*yardım et*" ("**help**", lit. "**do help**"); MWE:NCOMP, e.g., "*köşe yazarı*" ("**columnist**", lit. "**corner writer**"); and MWE:PROVERB for proverbs, as well as named entities grouped under MWE:ENAMEX / NUMEX / TIMEX, using the MUC nomenclature [18].

The integration of these types is expected to help the application of the treebanks to a number of additional high-level NLP tasks, such as transfer-based machine translation and semantic parsing.

4.2. Universal Dependencies

The UD project [19] has volunteer researchers from all over the world collaborating to make the largest multilingual collection of dependency treebanks to date. Besides providing more treebanks for UD and expanding its reach to more languages, UD contributors actively discuss data representation as well as morphological and syntactic annotation. UD annotation standards are revised with each new iteration in order to achieve treebanks

Table 1. Multiword expression subtypes.

Dependency relation	Description
mwe:comp	Idiomatic compounds with a nominal head
mwe:conj	Fixed compounds that behave like conjunctions
mwe:dup	Adverbials formed by reduplication
mwe:enamex:loc	enamex named entities referring to a location
mwe:enamex:org	enamex named entities referring to an organization
mwe:enamex:pers	enamex named entities referring to a person
mwe:formex	Formulaic expressions
mwe:idex	Idiomatic compounds with a finite verbal head
mwe:lvc	Light verb constructions
mwe:ncomp	Fossilized noun–noun compounds and other names
mwe:numex	numex named entities referring to generic numbers
mwe:numex:money	numex named entities referring to currency
mwe:numex:pct	numex named entities referring to a percentage
mwe:proverb	Proverbs
mwe:simex	Simile expressions with an idiomatic sense
mwe:timex:date	timex named expressions referring to a date
mwe:timex:time	timex named expressions referring to the time

that are more robust, successful, and balanced in terms of the tradeoff between computational tractability and linguistic correctness.

As contributors to the Turkish branch of the UD project, we have maintained and updated the IMST–UD Treebank since UD version 1.3. The conversion procedure of the treebank was partially automated but also involved a great deal of manual correction and reannotation in tokenization, morphology, and syntax. The example given in Figure 1 shows the extent to which the original and the UD annotation frameworks differ in tokenization, morphology, and syntax, demonstrating the need for manual intervention. The full conversion procedure can be seen in [15].

At the time of this writing, the most recent UD release was version 2.0 with 70 treebanks and 50 languages, including our IMST–UD Treebank representing the Turkish language. This version is a milestone because of its major leap from the last version, containing a number of radical changes and improvements in both annotation schemes and general data organization to accommodate a more diverse set of studies. In the course of this process, the Turkish language has evidently led the way for many of the adjustments made in annotation schemes that universally apply to all of the UD languages.

The fine-grained MWE annotations fused into the dependency layer described earlier in Section 3.2 have been fully utilized in the transition to UD v2.0. With their help, the IMST–UD Treebank was remapped to contain more precise compound annotations. The most recent version of the IMST–UD Treebank for UD v2.0 was made available in the official LINDAT repository for UD in March 2017, along with other treebanks.

Unlike well-edited treebanks, IWT contains a mix of noncanonical informal discourse and web jargon. This language is radically different from that of IMST [12] and therefore requires significantly different mapping processes. As a result, it was not trivial to map IWT to the UD standard using the same procedure. In

addition, at the time of UD v2.0, the IWT–UD Treebank was left for a future release. One of our most recent contributions is the mapping of IWT into the UD standard for the first time, creating IWT–UD as a candidate for the second Turkish UD treebank. Furthermore, the deep dependency layer presented in Section 3.1 is suitable for conversion to the enhanced dependency representation introduced with UD v2.0. The IWT–UD Treebank and the enhanced dependency layers for both Turkish UD treebanks are scheduled to be included in the next UD release.

5. Evaluation and discussion

This section contains our analyses of the four corpora described earlier in this article: the original IMST and IWT, as well as the UD-compliant IMST–UD and IWT–UD Treebanks. In Section 4.1, we first present some statistical figures extracted from each of the treebanks side by side to facilitate comparison. Later, in Section 4.2, we describe our empirical parsing tests along with preliminary information regarding the learning, parsing, and evaluation systems used and briefly discuss their results.

5.1. Statistics

We present a general breakdown of the sentences, tokens, and dependencies that constitute our treebanks in Table 2. Both treebanks demonstrate a slight increase in word counts but a marked decrease in token and dependency counts after the transition to UD. This is due to the UD approach to tokenization, which is significantly different from the inflectional group (IG) formalism (representation of subword units) used in the original treebanks [15]. Besides this, the distribution of nonprojective dependencies (overlapping dependency arcs) appears to be roughly equal in the original and UD versions of the treebanks, while the effect on average dependency distance (the average number of words between the dependent and the head in sentence order among all dependencies) seems ambiguous. The UD framework is more elaborate in its tag sets, although the original framework has a higher unique dependency relation count due to the MWE labels (Table 1).

Table 2. Comparative statistics for the IMST and IWT Treebanks.

	IMST	IMST–UD	IWT	IWT–UD
Sentences	5635	5635	5009	5009
(Orthographic) Words	56,422	58,085	43,191	44,463
(Syntactic) Tokens	63,066	58,146	47,226	44,545
Tokens w/o Deep Deps	61,585 (97.6%)	58,146	46,080 (97.6%)	44,545
Tokens with Deep Deps	1481 (2.4%)	–	1144 (2.4%)	–
Dependencies	64,812	58,146	48,497	44,545
Surface Deps (excl. deriv)	56,424	58,146	43,192	44,545
Surface Deps (incl. deriv)	63,066 (97.3%)	58,146	47,226 (97.4%)	44,545
Deep Deps	1746 (2.7%)	—	1271 (2.6%)	—
Projective Deps	62,831 (96.9%)	56,472 (97.1%)	47,278 (97.5%)	43,855 (97.5%)
Nonprojective Deps	1981 (3.1%)	1674 (2.9%)	1219 (2.5%)	690 (2.5%)
Average Dep. Distance	2.92	3.17	2.58	2.55
(Unique) Parts of Speech	11	14	11	15
(Unique) Morph. Features	47	74	46	64
(Unique) Dep. Relations	33	29	32	28

The distributions of parts of speech and dependency relations over all tokens are provided at http://tools.nlp.itu.edu.tr/Datasets. For explanations of these tags, please visit http://tools.nlp.itu.edu.tr (for the original framework) and http://universaldependencies.org (for the UD framework). The domain differences between IMST and IWT are quite evident from the data presented here. IWT has a significantly higher percentage of interjections and vocatives (cf. INTERJ/vocative in the original and INTJ/discourse in UD). Additionally, the web-crawled sentences featured in IWT seem to feature a higher concentration of determiners (cf. DET/determiner in the original and DET/det in UD), whereas the usage of punctuation seems to be significantly less frequent (cf. PUNC/punctuation in the original and PUNCT/punct in UD), although this is partially compensated by the usage of symbols such as emoticons (cf. SYM in UD).

5.2. Parsing Tests

The parsing scores presented in this section are obtained from applying tenfold cross-validation on each of the corpora. For syntactic parsing, we use MaltParser [28], a datadriven syntactic parser with a support vector machine infrastructure for statistical machine learning, following the setup in [9,15]. The parameters of the cited parsing setup are available for the purposes of replication. We eliminate nonprojective sentences from all training sets and exclude dependencies with the relation derıv in evaluating the accuracy of the prediction as is customary [8,9,15]. The relation derıv is a dummy relation used in IMST and IWT to denote intratoken relations between syntactic words, following the IG formalism. As required by the UD standard, word segmentation was done in a different way for IMST–UD and IWT–UD Treebanks, using a variety of dependency relations such as case and cop. In comparison, derıv relations are trivial for a parser to assign, and so they are not considered meaningful dependencies and are excluded when calculating accuracy scores. As deep dependencies are not supported in learning by the inherited parsing setup, we also ignore the deep dependency layer and run our tests only on surface dependencies.

The overall parsing scores obtained from cross-validation are given in the first part of Table 3. In this table, we include both labeled and unlabeled attachment scores for IMST and IWT as well as for their UD counterparts. There seems to be a slight improvement (Table 3, rows 6 and 9) in both attachment scores on the original versions of the treebanks since their latest evaluation [9,12] (75.3%→75.4% labeled and 83.7%→83.8% unlabeled for IMST, 79.7%→80.5% labeled and 87.5%→87.8% unlabeled for IWT), but a marked decrease (Table 3, row 2) on the UD version(s) since the release of IMST–UD [15] (77.1%→70.5% labeled and 83.8%→78.5% unlabeled) (the cross-validation scores given for the IWT–UD Treebank in Table 3 are from the very first evaluation of the treebank, so they cannot be compared with previous evaluations). We believe this is largely due to the change in the annotation schemes of conjunctions and punctuation, as they were trivial in previous UD versions.

Finally, the second part of Table 3 compares the parsing performances on three versions of the IMST and IWT that differ in terms of their MWE representations. The first version (mwe|original) indicates the original versions of the treebanks. In the second version (mwe|simplified), all subtypes of MWEs (e.g., mwe:idex, mwe:lvc) were replaced by the generic tag mwe. For the third version (mwe|none), all MWE relations were automatically substituted by the underlying syntactic relations, as in [25]. We interpret the results as follows: having subtypes for MWEs in the dependency layer provides a highly expressive label set at the expense of a minor computational load for the parser, but the annotation of MWEs in the first place comes at a steep cost in parsing. Even though parsing scores get significantly higher with less MWE annotation, we still opt for keeping them in place in our treebanks for the application potential argued in Section 3.2. Moreover, it is easy to convert MWE labels back to syntactic relations through a preprocessing stage, but it would not have been

Table 3. Labeled (LAS) and unlabeled (UAS) attachment scores.

		LAS	UAS
IMST		75.0 ± 0.2%	83.7 ± 0.3%
IMST–UD		70.5 ± 0.2%	78.5 ± 0.3%
IWT		79.8 ± 0.3%	87.7 ± 0.2%
IWT–UD		76.0 ± 0.4%	82.5 ± 0.4%
IMST	mwe | original	75.0 ± 0.2%	83.7 ± 0.3%
	mwe | simplified	75.4 ± 0.2%	83.8 ± 0.2%
	mwe | none	77.8 ± 0.2%	84.0 ± 0.2%
IWT	mwe | original	79.8 ± 0.3%	87.7 ± 0.2%
	mwe | simplified	80.5 ± 0.2%	87.8 ± 0.2%
	mwe | none	82.8 ± 0.2%	87.9 ± 0.1%

possible to automatically detect MWEs with gold-standard quality or assign them subtypes if the annotations were removed.

6. Conclusion

In this article, we introduced the final stable versions of our Turkish dependency treebanks (IMST and IWT). The treebanks and their most recent annotation guidelines are available for researchers at http://tools.nlp.itu.edu.tr/datasets. We described our critical contributions to these treebanks along with relevant evaluations and discussions. We also provided baseline parsing scores on the final versions of the treebanks. Our efforts brought about the long-overdue introduction of separate deep dependency layers and significantly more detailed multiword expression annotations in both treebanks. Furthermore, we converted both treebanks into the latest UD v2.0 standard. Finally, we presented a comprehensive set of statistics on the latest versions of both treebanks in comparison with their UD counterparts. We believe that the experimental figures and critical discussions presented in this article can serve as a useful resource for anyone who would employ IMST and IWT Treebanks in future studies.

References

[1] Kübler S, McDonald R, Nivre J. Dependency parsing. In: Heinz J, editor. Synthesis Lectures on Human Language Technologies. San Rafael, CA, USA: Morgan & Claypool, 2009.

[2] Percival WK. Reflections on the history of dependency notions in linguistics. Hist Ling 1990; 17: 29-47.

[3] Tesnière L. Éléments de Syntaxe Structurale. Paris, France: Éditions Klinksieck, 1959 (in French).

[4] Atalay NB, Oflazer K, Say B. The annotation process in the Turkish Treebank. In: 4th International Workshop on Linguistically Interpreted Corpora; 13–14 April 2003; Budapest, Hungary.

[5] Oflazer K, Say B, Hakkani-Tür DZ, Tür G. Building a Turkish treebank. In: Abeille A, editor. Building and Exploiting Syntactically-Annotated Corpora. Dordrecht, the Netherlands: Kluwer Academic Publishers, 2003.

[6] Buchholz S, Marsi E. CoNLL-X Shared Task on multilingual dependency parsing. In: 10th Conference on Computational Natural Language Learning; 8–9 June 2006; New York, NY, USA. New York, NY, USA: ACL. pp. 149-164.

[7] Eryiğit G. Dependency parsing of Turkish. PhD, İstanbul Technical University, İstanbul, Turkey, 2006.

[8] Eryiğit G, Nivre J, Oflazer K. Dependency parsing of Turkish. Comput Linguist 2008; 34: 357-389.

[9] Sulubacak U, Pamay T, Eryiğit G. IMST: A revisited Turkish dependency treebank. In: 1st International Conference on Computational Turkish Linguistics; 3 April 2016; Konya, Turkey.

[10] Eryiğit G. ITU Turkish NLP web service. In: 14th Conference of the European Chapter of the Association for Computational Linguistics; 2014; Gothenburg, Sweden. pp. 1-8

[11] Şeker GA, Eryiğit G. Extending a CRF-based named entity recognition model for Turkish well formed text and user generated content. Semant Web 2017; 8: 625-642.

[12] Pamay T, Sulubacak U, Torunoğlu-Selamet D, Eryiğit G. The annotation process of the ITU Web Treebank. In: Proceedings of the 9th Linguistic Annotation Workshop; 5 June 2015; Denver, CO, USA.

[13] Bies A, Mott J, Warner C, Kulick S. English Web Treebank. Philadelphia, PA, USA: LDC, 2012.

[14] Seddah D, Sagot B, Candito M, Mouilleron V, Combet V. The French social media bank: a treebank of noisy user generated content. In: 24th International Conference on Computational Linguistics; December 2012; Mumbai, India.

[15] Sulubacak U, Gökırmak M, Tyers F, Çöltekin Ç, Nivre J, Eryiğit G. Universal dependencies for Turkish. In: 26th International Conference on Computational Linguistics; 11–17 December 2016; Osaka, Japan. pp. 3444-3454.

[16] Schuster S, Manning CD. Enhanced English universal dependencies: an improved representation for natural language understanding tasks. In: 10th International Conference on Language Resources and Evaluation; 23–28 May 2016; Portorož, Slovenia.

[17] Rosén V, Losnegaard GS, De Smedt K, Bejcek E, Savary A, Przepiórkowski A, Osenova P, Mititelu VB. A survey of multiword expressions in treebanks. In: 14th International Workshop on Treebanks & Linguistics; December 2015; Warsaw, Poland.

[18] Sundheim B. Overview of results of the MUC-6 evaluation. In: 6th Message Understanding Conference; 6–8 November 1995; Columbia, MD, USA.

[19] Nivre J, de Marneffe MC, Ginter F, Goldberg Y, Hajič J, Manning CD, McDonald R, Petrov S, Pyysalo S, Silveira N et al. Universal Dependencies v1: A multilingual treebank collection. In: 10th International Conference on Language Resources and Evaluation; May 2016; Paris, France.

[20] Sulubacak U. Improving statistical dependency parsing performance in Turkish by use of a new annotation scheme. MSc, İstanbul Technical University, İstanbul, Turkey, 2015 (in Turkish with a summary in English).

[21] Baldwin T, Kim SN. Multiword expressions. In: Indurkhya N, Damerau FJ, editors. Handbook of Natural Language Processing. 2nd ed. Boca Raton, FL, USA: Chapman and Hall/CRC; 2010. pp. 267-292.

[22] Sang EFTK, De Meulder F. Introduction to the CoNLL-2003 shared task: language-independent named entity recognition. In: 7th Conference on Natural Language Learning; 2003; Edmonton, Canada. New York, NY, USA: ACL. pp. 142-147.

[23] Çelikkaya G, Torunoğlu D, Eryiğit G. Named entity recognition on real data: a preliminary investigation for Turkish. In: 7th Annual Conference on Application of Information and Communication Technologies; 23–25 October 2013; Baku, Azerbaijan. New York, NY, USA: IEEE. pp. 1-5.

24 Eryiğit G, Adalı K, Torunoğlu-Selamet D, Sulubacak U, Pamay T. Annotation and extraction of multiword expressions in Turkish treebanks. In: 11th Workshop on Multiword Expressions; 4 June 2015; Denver, CO, USA.

[24] Eryiğit G, İlbay T, Can OA. Multiword expressions in statistical dependency parsing. In: 2nd Workshop on Statistical Parsing of Morphologically-Rich Languages; 6 October 2011; Dublin, Ireland. New York, NY, USA: ACL. pp. 44-55

[25] Savary A, Sailer M, Parmentier Y, Rosner M, Rosén V, Przepiórkowski A, Krstev C, Vincze V, Wójtowicz B, Losnegaard GS et al. PARSEME–PARSing and multiword expressions within a European multilingual network. In: 7th Annual Language and Technology Conference; 2015; Poznań, Poland.

[26] Adalı K, Dinç T, Gökırmak M, Eryiğit G. Comprehensive annotation of multiword expressions for Turkish. In: 1st International Conference on Computational Turkish Linguistics; 3 April 2016; Konya, Turkey. pp. 60-66.

[27] Nivre J, Hall J, Nilsson J, Chanev A, Eryiğit G, Kübler S, Marinov S, Marsi E. MaltParser: A language-independent system for data-driven dependency parsing. Nat Lang Eng 2007; 13: 95-135.